Estimation of Physical Layer Scrambling Code Sequence of DVB-S2

Hao Wu, Hui Xie, Zhi-Tao Huang, and Yi-Yu Zhou

In this letter, the problem of estimating the physical layer (PL) scrambling code sequence of DVB-S2 is studied. We present the first ever scheme to estimate the scrambling sequence. The scheme is based on hypothesis testing. By analyzing the PL scrambling process, we construct a new sequence equivalent to the scrambling sequence. We then use hypothesis testing to estimate the new sequence. The threshold for the hypothesis testing is also discussed. The experiment results show that the performance of our estimation scheme can work even under high BER.

Keywords: Hypothesis testing, DVB-S2, scrambling sequence.

I. Introduction

In the DVB-S2 system, the scrambling code sequence is unequivocally associated with each satellite operator or satellite or transponder [1]. For non-cooperative communication, we can identify the transmitter from the scrambling code sequence. Only after we obtain the scrambling code sequence can the physical layer frame (PLFrame) be recovered. Hence, estimation of the scrambling code sequence is very significant for non-cooperative communication. At present, research about scrambling code mainly focuses on the estimation of the generator polynomial. For example, when some input and scrambled bits are known, the Berlekamp-Massey (BM) algorithm can be used to reconstruct the feedback polynomial [2]. If only the scrambled bits are known, Cluzeau's proposed algorithm can be used [3], [4]. The BM algorithm is based on the scrambling code sequence, but this algorithm does not provide the solution obtaining the scrambling code sequence. Cluzeau's algorithm is only applicable to a single-channel scrambled code sequence. Regarding the scrambled code sequence of DVB-S2, the scrambling code sequence is separated into the in-phase channel and the quadrature channel. Hence, Cluzeau's algorithm is inapplicable to DVB-S2. The problem of estimating the scrambling code sequence, especially for DVB-S2, has yet to be solved. In this letter, we propose the first ever scheme to estimate the PL scrambling code sequence of DVB-S2.

Manuscript received Sept. 12, 2013; revised Oct. 2, 2013; accepted Oct. 9, 2013.

Hao Wu (phone: +86 073184573489, wuhao04.thu@gmail.com), Hui Xie (xiehui2005@gmail.com), Zhi-Tao Huang (taldcn@yahoo.com.cn) and Yi-Yu Zhou (zhouyiyu@sohu.com) are with College of Electronic Science and Engineering, National University of Defense Technology, Changsha, Hunan, P.R. China.

II. Physical Layer Scrambling of DVB-S2

Prior to modulation, each PLFrame of DVB-S2, excluding the physical layer header (PLHeader), shall be randomized for energy by multiplying the $(I+jQ)$ samples by a complex randomization sequence (C_I+jC_Q), as shown in Fig. 1.

The scrambling sequence is constructed by combining two real *m*-sequences into a complex sequence. The resulting sequence thus constitutes segments of a set in a Gold sequence. Let x and y be the two sequences. The x sequence is constructed using the primitive polynomial $1+x^7+x^{18}$. The y sequence is constructed using the polynomial $1+y^5+y^7+y^{10}+y^{18}$.

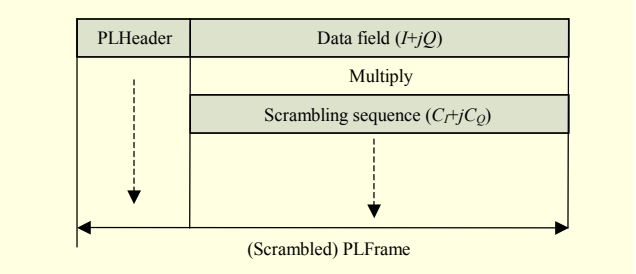

Fig. 1. PL scrambling of DVB-S2.

Table 1. Relationship between scrambled sequence and scrambling sequence.

R_n	C_I	C_Q	$I_{\text{scrambled}}$	$Q_{\text{scrambled}}$	$I_{\text{scrambled}} * Q_{\text{scrambled}}$	$z_n(i)$
0	1	0	I	Q	IQ	0
1	0	1	$-Q$	I	$-IQ$	1
2	–1	0	$-I$	$-Q$	IQ	0
3	0	–1	Q	$-I$	$-IQ$	1

The construction of *m*-sequences x and y is as follows [1]. The initial conditions are

$$x(0)=1,\ x(1)=x(2)=\ldots=x(16)=x(17)=0 \quad (1)$$

and

$$y(0)=y(1)=\ldots=y(16)=y(17)=1. \quad (2)$$

The recursive definition of subsequent symbols is

$$x(i+18)=\mathrm{mod}\left(x(i+7)+x(i),2\right), \quad (3)$$

$$y(i+18)=\mathrm{mod}\left(y(i+10)+y(i+7)+y(i+5)+y(i),2\right), \quad (4)$$

where mod(,) is the modulus operator. The *n*-th Gold code sequence z_n is then defined as [1]

$$z_n(i)=\mathrm{mod}\left(\left[x\left(\mathrm{mod}\left((i+n),2^{18}-1\right)\right)+y(i)\right],2\right). \quad (5)$$

The binary sequence is converted to integer-valued sequence R_n by the following transformation [1]:

$$R_n(i)=2z_n\left(\mathrm{mod}\left((i+131072),2^{18}-1\right)\right)+z_n(i). \quad (6)$$

Finally, the *n*-th complex scrambling code sequence, $C_I(i)+jC_Q(i)$, is defined as [1]

$$C_I(i)+jC_Q(i)=\exp\left(jR_n(i)\pi/2\right). \quad (7)$$

The relationship between the above-listed sequences is shown in Table 1.

III. Estimation of Scrambling Code Sequence

In this section, we assume that bit synchronization and frame synchronization are achieved. Therefore, we must only focus on the baseband signal. From Table 1, it can be observed that the $I_{\text{scrambled}}$ sequence and $Q_{\text{scrambled}}$ sequence are both a mix of sequences I and Q. We cannot obtain the Gold code sequence $z_n(i)$ from either $I_{\text{scrambled}}$ or $Q_{\text{scrambled}}$. To recover the scrambling code sequence, we construct a new sequence, F, by multiplying $I_{\text{scrambled}}$ and $Q_{\text{scrambled}}$.

$$F=I_{\text{scrambled}} * Q_{\text{scrambled}}=S*(I*Q), \quad (8)$$

where $S=(s_1,s_2,\ldots,s_N)$ is an unknown sequence equivalent to the scrambling sequence obtained in the following and N is the PLFrame length.

We regard the product of the *I*-*Q* sequence to be the input bits $C=(c_1,c_2,\ldots,c_N)$.

$$C=I*Q. \quad (9)$$

Rewrite (8) as

$$F=S*C. \quad (10)$$

From Table 1, we get

$$\begin{cases} s_i=-1, & i\in\{i\mid \mathrm{mod}(R_n(i),2)=1\},\\ s_i=1, & i\in\{i\mid \mathrm{mod}(R_n(i),2)=0\}. \end{cases} \quad (11)$$

From (6), we can write

$$z_n(i)=\mathrm{mod}\left(R_n(i),2\right). \quad (12)$$

Comparing (11) and (12), we obtain

$$z_n(i)=(1-s_i)/2. \quad (13)$$

The problem of estimating the scrambling sequence z_n now becomes the problem of estimating sequence S.

Assuming the biased memoryless source sequence to be b_i [2],

$$P[b_i=1]=1/2+\varepsilon_0,\quad \varepsilon_0\neq 0. \quad (14)$$

Then,

$$P[b_{2k-1}b_{2k}=1]=1/2+2\varepsilon_0^2. \quad (15)$$

Rewriting $2\varepsilon_0^2$ as ε, according to (9) and (15), we get

$$P[c_i=1]=1/2+\varepsilon,\ \varepsilon\neq 0. \quad (16)$$

Each element of the v-th PLFrame of the PL scrambled sequence can be expressed as $F_v^{def}=\left[f_{v,0},f_{v,1},\ldots,f_{v,j},\ldots,f_{v,N-1}\right]$, where v = 0, 1, ..., K–1 and K is the PLFrame number, $f_{v,j}=c_{v,j}s_j$. We use the sum of $f_{v,j}$ to estimate the PL scrambling code sequence.

$$x_j=\sum_{v=1}^{K}f_{v,j}=\sum_{v=1}^{K}c_{v,j}s_j=s_j\sum_{v=1}^{K}c_{v,j}. \quad (17)$$

The log likelihood ratio of $c_{v,j}$ can be facilitated by [5]

$$\Lambda_C(c_{v,j})=\ln\frac{P[c_{v,j}=1]}{P[c_{v,j}=-1]}=\ln\frac{1/2+\varepsilon}{1/2-\varepsilon}. \quad (18)$$

Then, we have

$$P_1=P[c_{v,j}=1]=\frac{e^{\Lambda_C(c_{v,j})}}{1+e^{\Lambda_C(c_{v,j})}}, \quad (19)$$

$$P_2 = P\left[c_{v,j} = -1\right] = \frac{1}{1+e^{\Lambda_C\left(c_{v,j}\right)}}. \tag{20}$$

The following section, we shall illustrate two different cases. The first case is $\varepsilon > 0$, assuming two hypothesizes,

$$\begin{aligned} H_0 &: s_j = -1, \\ H_1 &: s_j = 1. \end{aligned} \tag{21}$$

We get the probability distribution function (PDF) of each hypothesis.

$$\begin{aligned} p\left(x_j; H_1\right) &= \frac{e^{\Lambda K/2}}{\left(1+e^{\Lambda}\right)^K}\begin{pmatrix} K \\ \left(K+x_j\right)/2 \end{pmatrix} e^{\Lambda x_j/2}, \\ x_j &= -K, -K+2, \ldots\ldots, K-2, K, \end{aligned} \tag{22}$$

where $\begin{pmatrix} K \\ \left(K+x_j\right)/2 \end{pmatrix}$ means the number of combinations of K taken $\left(K+x_j\right)/2$ at a time.

$$\begin{aligned} p\left(x_j; H_0\right) &= \frac{e^{\Lambda K/2}}{\left(1+e^{\Lambda}\right)^K}\begin{pmatrix} K \\ \left(K-x_j\right)/2 \end{pmatrix} e^{-\Lambda x_j/2}, \\ x_j &= -K, -K+2, \ldots\ldots, K-2, K. \end{aligned} \tag{23}$$

If we use the Neyman-Pearson theorem [6], giving the probability of false alam (PFA) β=10^{-3}, then

$$P_{FA}(\gamma) = \int_{\{x_j : L(x_j) > \gamma\}} p\left(x_j; H_0\right) dx_j = \beta. \tag{24}$$

The likelihood ratio is

$$L\left(x_j\right) = p\left(x_j; H_1\right) \Big/ p\left(x_j; H_0\right) = e^{\Lambda x_j}. \tag{25}$$

Then,

$$P_{FA}(\gamma) = \int_{\{x_j : e^{\Lambda x_j} > \gamma\}} p\left(x_j; H_0\right) dx_j = 10^{-3}. \tag{26}$$

From (23) and (26), we obtain

$$\gamma = 37.0427. \tag{27}$$

According to (25), we can get

$$x_{th} = \ln \gamma / \Lambda = 18. \tag{28}$$

The probability of detection (PD) is [6]

$$P_D\left(x_{th}\right) = \int_{\{x_j > x_{th}\}} p\left(x_j; H_1\right) dx_j = 0.1343. \tag{29}$$

The detection performance is poor. We use the difference of the PD and the PFA to determine the detection threshold for hypothesis testing. According to (25) and (26), we can write

$$\begin{aligned} P_{FA}(\gamma) &= \int_{\{x_j : L(x_j) > \gamma\}} p\left(x_j; H_0\right) dx_j \\ &= \int_{\left\{x_j > \frac{\log_e \gamma}{\Lambda}\right\}} p\left(x_j; H_0\right) dx_j. \end{aligned} \tag{30}$$

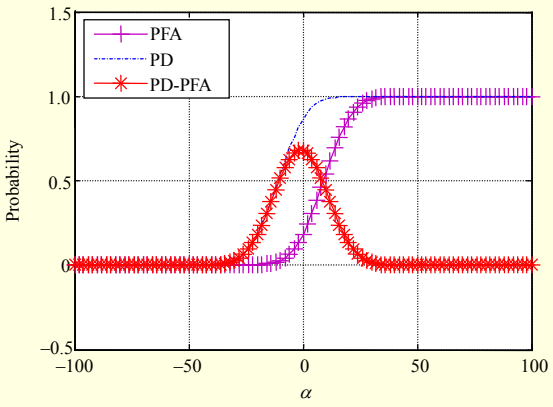

Fig. 2. PD, PFA, and their difference.

Rewriting γ' as $\frac{\log_e \gamma}{\Lambda}$, we get

$$P_{FA}(\gamma') = \int_{\{x_j > \gamma'\}} p\left(x_j; H_0\right) dx_j. \tag{31}$$

Unify the variable in (29) and (31) as α and calculate the difference of the PD and the PFA.

$$P_{D\_FA}(\alpha) = P_D(\alpha) - P_{FA}(\alpha), \tag{32}$$

as shown in Fig. 2.

Then, we calculate the threshold x_{th} from the following equality instead of from (26):

$$x_{th} = \left\{\alpha \mid P_{D\_FA}(\alpha) = \max\left(P_D(\alpha) - P_{FA}(\alpha)\right)\right\}. \tag{33}$$

Using the new threshold, we get the estimation rule.

$$\begin{cases} \hat{s}_j = -1, & x_j \le x_{th}, \\ \hat{s}_j = 1, & x_j > x_{th}. \end{cases} \tag{34}$$

When $\varepsilon < 0$, then, (22) and (23) can be rewritten as

$$\begin{aligned} p\left(x_j; H_1\right) &= \frac{e^{\Lambda K/2}}{\left(1+e^{\Lambda}\right)^K}\begin{pmatrix} K \\ \left(K-x_j\right)/2 \end{pmatrix} e^{-\Lambda x_j/2}, \\ x_j &= -K, -K+2, \ldots\ldots, K-2, K, \end{aligned} \tag{35}$$

$$\begin{aligned} p\left(x_j; H_0\right) &= \frac{e^{\Lambda K/2}}{\left(1+e^{\Lambda}\right)^K}\begin{pmatrix} K \\ \left(K+x_j\right)/2 \end{pmatrix} e^{\Lambda x_j/2}, \\ x_j &= -K, -K+2, \ldots\ldots, K-2, K. \end{aligned} \tag{36}$$

We get the following estimation rule:

$$\begin{cases} \hat{s}_j = 1, & x_j \le x_{th}, \\ \hat{s}_j = -1, & x_j > x_{th}. \end{cases} \tag{37}$$

IV. Numerical Results

To validate the estimation scheme performance, simulation

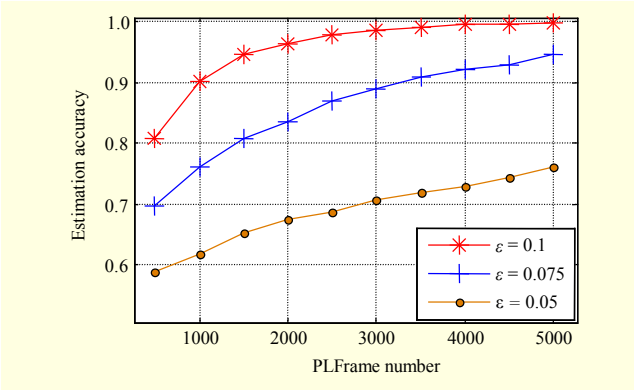

Fig. 3. Performance of estimation under different ε.

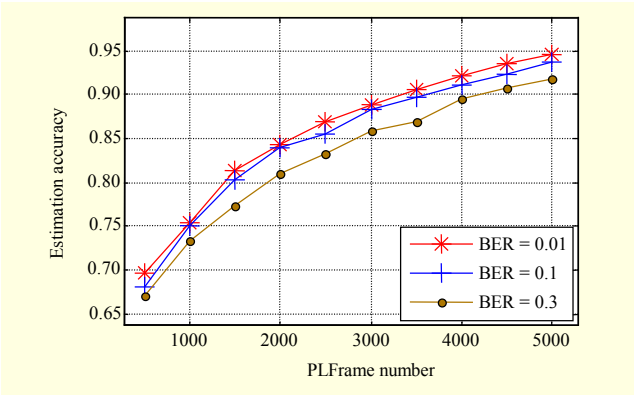

Fig. 4. Performance of estimation under different BER.

experiments are conducted under different situations. Firstly, the BER of the PL scrambled sequence is 0.1. As illustrated in Fig. 3, the performance of estimation decreases when ε lessens. In other words, we need more PLFrames to estimate the scrambling code sequence.

Secondly, when ε is 0.075, the BERs of the PL scrambled sequence are chosen as variable values. Figure 4 shows that the estimation accuracy slightly decreases as the BER increases. When the frame number is larger than 2,000, the estimation accuracy is still bigger than 0.8, even when the BER=0.3. Although the estimation accuracy is less than 0.7 when the frame number is 500, it is enough for us to obtain the scrambling sequence. According to [1], sequences x and y are invariable. If we are able to obtain the order n of Gold code sequence Zn, as in (5), we can then obtain the whole scrambling code sequence. From (5) and (13), we can get $\hat{z}_n(i)=(1-\hat{s}_j)/2$, $\hat{x}(\mathrm{mod}((i+n),2^{18}-1))=\mathrm{mod}(\hat{z}_n(i)+y(i),2)$. Then, we calculate the cross correlation of $x(i)$ and $\hat{x}(\mathrm{mod}((i+n),2^{18}-1))$. The value of the x coordinate corresponding to the maximum is of order n.

V. Conclusion

A robust estimation scheme of the PL scrambling code sequence was presented in this letter. The scheme is based on the hypothesis testing of a new estimation variable. As for the judgment threshold, we proposed a new get method instead of using the Neyman-Pearson theorem. We considered the probability of detection and the probability of false alarm. The simulation results show that the performance of our estimation scheme can work even under high BER.

References

[1] ETSI EN 302-307, "Digital Video Broadcasting (DVB) — Second Generation Framing Structure, Channel Coding and Modulation Systems for Broadcasting, Interactive Services, News Gathering, and Other Broadband Satellite Applications," V1.1.2, July 2006.

[2] J.L. Massey, "Shift-Register Synthesis and BCH Decoding," *IEEE Trans. Inf. Theory*, vol. 15, no. 1, Jan. 1969, pp. 122-127.

[3] M. Cluzeau, "Reconstruction of a Linear Scrambler," *IEEE Trans. Comput.*, vol. 56, no. 9, Sept. 2007, pp. 1283-1291.

[4] X.-B. Liu et al., "Reconstructing a Linear Scrambler with Improved Detection Capability and in the Presence of Noise," *IEEE Trans. Inf. Foren. Sec.*, vol. 7, no. 1, Feb. 2012, pp. 208-218.

[5] J. Hagenauer, E. Offer, and L. Papke, "Iterative Decoding of Binary Block and Convolutional Codes," *IEEE Trans. Inf. Theory*, vol. 42, no. 2, Mar. 1996, pp. 429-445.

[6] S.M. Kay, *Fundamentals of Statistical Signal Processing, Volume II: Detection Theory*, Upper Saddle River, NJ: Prentice Hall PTR, 1998.